Graphics asset production

Project manager's responsibilities

■ To assemble the necessary graphics team

■ To ensure that the project definition is in a form that adequately briefs the graphics team

■ To monitor the development of the graphics, and liaise between the client and the team over changes and misunderstandings

■ To make sure that 'proofreading', the checking of any written material in the graphics, is carried out

■ To make sure that the source graphics are archived at the highest reasonable resolution, no matter how they are displayed in the application

■ To understand the processes involved in producing this kind of asset

■ A picture is worth a thousand words

No matter how innovative your interface design, how sophisticated your programming or use of HTML, the public face of your web page, your inter-active television programme or multimedia application and the major make-or-break factor is going to be your graphics and design. (Possibly not a big issue for a WAP site at the moment however.) Your potential customer will feel able to make an instant judgement, for better or for worse, on the basis of that first impression on the screen.

Just as with software, the creation and manipulation of graphics is a com-plex business, and the purpose of this chapter is to provide some general background to the techniques and systems that you might wish to use, and to outline some of the problems that you might come across.

The two key parts of managing the production of graphics are the defini-tion of the task and the selection of the personnel. There should be some definition of the approach to graphics in the project plan. In some cases a graphics artist or art director will have taken part in the creative definition of the application. On the basis of the graphic requirements a team will be chosen. It may be that you are working in an organization that has graphics artists on the staff or a pool of freelancers. This situation is more common for graphics and programming than for sound or video. In this case you might find that the project plan evolved to fit in some way with the abilities of a particular graphics person.

There are many kinds of process that go to make up artwork, and after a brief explanation of terminology for colour, this chapter will look at the dif-ferences between them and how the way they are made may impact upon the management of a project. This will apply to two stages, because you should work with your images in the highest quality possible and then con-vert down if necessary to the standard required for delivery. Archiving should also be done so as to include the highest-quality versions.

■ Colour

A colour image on a computer or television screen – whether it is a cathode ray tube (CRT), liquid crystal (LCD) or gas plasma – is made up of red, green and blue dots or lines, which are close enough together to blend into each other when seen from a working distance. Red, green and blue (R, G and B) are what are called the primary colours for any colour displayed with light. You will find an example on the website of how a colour image is built in this way. (For pigments, such as paint and printing ink, the primaries are red, yellow and blue. For practical reasons printers work with secondary colours – cyan, magenta and yellow – to which black ink – called K – is usually added to avoid the muddy effects of trying to make black from the secondaries. This is known as CMYK.)

For this reason we refer to a colour image on the screen as an RGB image. (RGB is one of a number of ways of defining a colour, other so-called colour spaces include CMYK and hue-saturation-brightness which is known as HSB but in this discussion we will stick with RGB.) For most practical purposes, almost any colour can be produced by mixing the three primaries, and in a full-colour image on the screen each picture element or pixel is built up from varying amounts of red, green and blue light. Shades of grey are produced by making the amount of red, green and blue in the pixel equal. For black there is no light, and for white the light is 'full on'. In digital terms there are 256 shades to each of red, green and blue in a full-colour 24-bit image.

With fewer than 24 bits there will be fewer colours, but it is possible that each of the colours that can be shown can be predefined from the full range, or palette, of 24 bits.

■ Drawings

This word has a double meaning. To the person in the street, drawing is taking a pencil and creating a freehand work of art. Drawing, like painting, is the essence of fine art.

In computer terms a drawing is an image that consists of distinct segments or shapes, called draw objects. It is sometimes referred to as line art but this term could be ambiguous, sometimes also referring to an image on the screen that is made up of only black and white pixels.

The classic draw objects are lines, rectangles and arcs together with a range of wavy lines called curves, of which Bézier curves are perhaps the
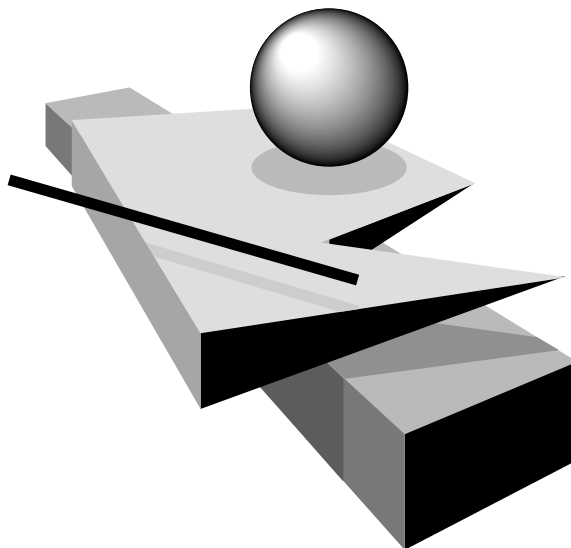
most familiar. Since the draw objects are based around lines they are sometimes called vector graphics. The important thing about a drawing is that the objects in it retain their separate identity, so that they can be manipulated independently of each other, even if, in the image as seen, the object is partially or completely hidden behind other objects.

Each object has characteristics, which can be changed. Drawing applications will allow you to rescale an object or a group of objects without a loss of quality. The width of a line, and the pattern with which it is drawn, called the pen, can be changed. If the object has an 'inside', it can be filled with a colour or a pattern. A polygonal shape or a curve can be altered by moving the corners or changing the controls on the curve.
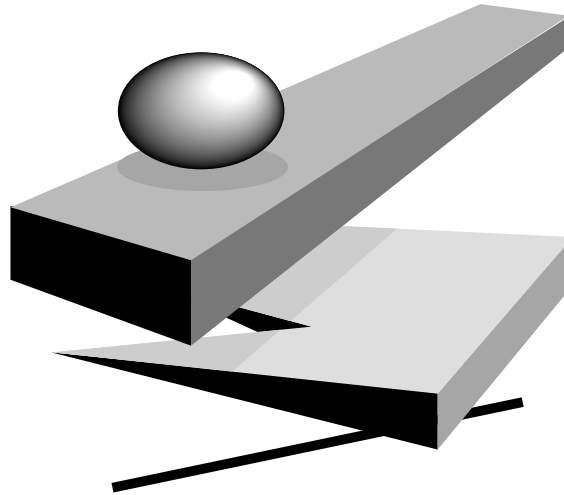
Changing one object has no substantial repercussions on the other objects. If the circle in the diagram is moved it would reveal the polygon behind it. The image becomes a pattern of dots on the monitor only when it is displayed or when you choose to 'fix' it. By definition the drawing does not have a scale, and can be expanded or shrunk or distorted as required with no ill effects, as seen above. An exception to this scale independence is the thickness of lines, since that tends to be fixed. As a result you might wish to increase line thickness as the size of the final drawing increases, and conversely reduce line thickness for a small drawing.

If changes are required, drawings are relatively easy to modify. Because it is based around these distinct and almost unconnected objects a drawing package is said to be object-oriented.

In some computing systems, particularly the Apple Macintosh, these draw objects are supported by the operating system at a low level, which makes it easy for different programs to work with them and even exchange them.

Although most graphics on the Web are bitmapped, there is a small but important use of draw objects in making small animation files. Flash is a particularly popular example of this, and by describing the objects that go to make up an animation, and describing how they move and change, Flash is able to produce sophisticated animations from a very small amount of data.

■ Bitmap or painting

In contrast a bitmap, sometimes referred to in computing as a painting (hence paint packages as opposed to drawing packages) or rasterized graphics, exists only as the patterns of dots. To continue the example above, if the ellipse moves it leaves a hole. If the bitmap expands, the dots get bigger and the resolution of the image becomes cruder.

The bitmap may start out as a set of draw objects. It may be created in a drawing package and then turned into a bitmap, 'fixing' the pattern of lines and shapes. This process is also known as rasterizing. The bitmap could have been scanned in from a photograph. Scanners are devices that take a 'real' image such as a print, a slide, a transparency or even a photographic negative and scan across the image, from side to side and top to bottom, to produce a stream of dot information about the image, which can be stored in a computer file and so make its way onto the screen.

Making changes to a bitmap, such as changing text in the image, could have serious implications because of the holes left when parts of the image are removed. One way around this problem is to use a paint package which allows control of layers of an image and the way they are combined, or composited, together.

JPEG and GIF are the two most common bitmap formats for graphics, and both are used extensively on the Web. On CD-ROMs, JPEGs are also common, although GIFs are unusual. BMP (Windows) and PICT (Mac) are platform-specific bitmap formats used on CD-ROMs. The older TIFF format is often now found in digital still cameras as an uncompressed format. The new graphics format PNG supports up to 48 bits of colour (16 bits each for R, G and B), which allows the look of the graphic to be adjusted as it is displayed to compensate for differences between different computers' graphics displays.

■ Composites

The addition of alpha channels and layers into paintings has helped to bring some of the advantages of draw objects to the world of the bitmap. Separate bitmap objects can be combined into a composite image. Even though changing a bitmap object itself is still difficult, the relationship between objects can easily be changed. Text (one object) over a photographic background

(another object) can easily be changed. Each component of the final image is known as a layer, and the alpha channel determines how the layers blend with each other.

The simplest alpha channel is transparency or a simple mask, so-called because it masks or hides part of the image. The GIF format allows for a colour to be transparent, which allows the background to show through, and helps the image to blend into the background of the web page.

A true alpha channel is a graduated mask. It is essentially a fourth channel after red, green and blue, and it determines how much of the RGB image layer is shown at any point of the composite image and how much of whatever is underneath shows through. For most of the image the mask would be 'white', showing that all the RGB was visible and none of the underlying layers. Outside the image it would be 'black', showing that none of the RGB was visible there but the underlying layers were fully visible. At the edges of the image, where it would be designed to blend into its surroundings, the alpha channel would gradually change from white to black, and the depth of the 'grey' at any point would determine the visibility and mixing at that point. TIFF and PNG formats support an alpha channel, as does the Macintosh PICT format.

In a montage, where there might be several small images grouped together over a background, each of the small images would have its own alpha channel and would exist in a separate layer. In this way changes can be made to each of these elements independently of the whole composite

image. A further refinement of this technique would allow layers to be completely transparent or translucent. A reflection in a window would be transparent, with the image behind the glass being opaque behind it.

As a result of using layers in this way an image becomes easier to change under some circumstances. Let's use the example where a final image consists of a photograph with some text superimposed on it. If there is no layering to the image in your archive then to change the text you will have to find the original image of the background and build the image again from scratch. With layers you can change the text, which will be in the foreground layer by itself, without having to touch the background. A recent innovation is to use a version of layers to hold changes to an image, so that the changes can be modified or even reversed.

Paint packages can be biased towards creation, retouching or both. For many graphic artists Adobe's PhotoShop defines the retouching package (although it is not alone), especially once it became available on the PC platform as well as on the Apple Macintosh. Recent versions of PhotoShop included Image Ready, which prepares graphics for the Web. An alternative web-oriented graphics package is Fireworks. In the Linux world there is an equivalent free package called the GIMP.

■ 3-D modelling and rendering

Quite often an application will have illustrations and montages that have a three-dimensional look. This may or may not include an animation such as a smooth movement of the viewpoint through a group of objects. 3-D packages are usually object-oriented like a drawing package. You can build a solid object using techniques analogous to building with bricks, to using a lathe to turn, or to extrude (like squeezing toothpaste). Just as two-dimensional draw objects once became part of low-level operating system tools (sometimes called primitives), in time three-dimensional objects are joining them. This makes the creation of 3-D objects easier and cheaper because, again, programs will be able to make use of the operating system's objects rather than having to define their own.

Often speed of display is more important than realism and it can be more convenient to work with a wireframe version of the three-dimensional shape. This can be very useful with a very complex picture but it takes some skill to visualize how the wireframe will look when it is fully rendered.

Once the scene is designed and the objects in it are built and positioned you can choose the surfaces for the objects. This can include texture mapping, where a bitmap is spread over the surface of an object, bump mapping, where the object is given a bumpy surface, and setting the reflectivity or opacity of the object.

Then the lighting conditions can be set up with a balance of ambient (general) lighting and spot and floodlights, and the viewpoint chosen. All these parts of the process are part of the modelling.

Finally the scene is rendered to produce the final image or images. Unlike other graphical processes, which work more or less in real time, rendering can be very slow. A single image can take minutes or even hours to be rendered as the computer works out the view (depending on the power of the computer being used of course) and if an animation is being produced this multiplies the time. To overcome this several computers are often used together to gain speed. The end result can be quite stunning, and can be genuinely referred to as photo-realistic. This is one reason why 3-D modelling is such a popular technique.

3-D images take a long time to render partly because of the time taken to work out the shadows and reflections in a scene. One very popular technique for determining reflections is ray tracing. In this case every ray of light in the view is traced as it 'bounces' from the light source, the objects in the scene and the viewer's eye. Ray tracing is very computationally intensive, and an alternative is environment mapping, where a view of 'the world as seen by the shiny object' is computed and wrapped around the object itself.

Three-dimensional modelling, like illustration and animation, is a special skill somewhat distinct from general graphics work, and you are likely to find artists who specialize in one or more of these areas.

A special case of 3-D modelling is VRML, Virtual Reality Modelling Language, which is available to be used on the Web to make virtual worlds that the viewer can navigate at will. The VRML data defines the environment, and this is modelled in real time by a plug-in to the viewer's browser. The process needs a lot of computing and so is usually somewhat clunky compared with a sophisticated 3-D rendered image, but the real-time response to movement is very engaging, and it is possible to provide simple shading to enhance the experience. VRML evolved to include animations and sound so that you can watch events unfold in front of you from whatever perspective you want but, as a web technology, it has had limited success.

A similar process to this kind of modelling is provided by QuickTime VR (QTVR), which is usually used to show the real world in a navigable way. The world is photographed as an interlinked series of 360° panoramas, allowing the user to look around and move from one panorama to another. QTVR can explore other worlds as well, since not only can the panoramas be computer-generated, but NASA published QTVR panoramas taken by a rover vehicle on Mars. QTVR is not the only such navigation software and with some it is possible to look around a whole spherical scene.

■ DPI and bit depths

In computing terms you define an image in a number of ways. For a bitmap the most obvious defining parameter is size. In printing, graphics artists are used to defining an image by its size in inches or centimetres, and then saying what the resolution is. This means that if you have a scanned image that

is 2 inches across, with a resolution of 300 lines (or dots) per inch, there are 600 dots across the image. Print is defined in this way because print has a fixed size: the size of the final image on the printed page. Computer displays are not necessarily like that, and television screens are definitely not like that. In these cases the size in inches is not important, but the size in pixels is. So the computer person will give the image size as 600 pixels across and not really be worried about its physical size in inches. Note that to a printer a screen is part of the process by which an image made of continuous gradations of colour, such as a photograph, is turned into something that an ink-based printing press can actually reproduce. This book uses the word 'screen' to mean the computer display.

Certainly a computer display has its equivalent of the printer's screen. Dots per inch are the usual way of defining a computer screen output as well. This is usually 72, 75, 80 or even as many as 120 dots per inch (dpi), and each dot is a pixel. For a computer display of 75 dpi, the 600-pixel image will be 8 inches across. You need to be clear in your mind about the relationship between the size of an image in pixels, in inches, and the connection between the two, which is its dpi. If the image on the screen is to be 2 inches across again, instead of 8, then it can be shrunk by half in a paint or photo-retouching package. (You can scale an image in a web browser when you display it of course, but that is inefficient and sometimes of poorer quality.)

There is a third dimension to computer screen images, which is the bit depth or colour depth. The number of bits in the screen display determines the number of colours or shades that can be shown. For a fixed size of image in pixels, the size of the file needed to store it doubles for most of the stages below.

- 1 bit gives you two colours (black and white).
- 2 bits give you four colours.
- 4 bits give you 16 colours.
- 8 bits give you 256 colours (GIFs are 8 bits or less).
- 16 bits give you 65,536 colours.
- 24 bits give you 16,777,216 colours (JPEGs are usually 24 bit but can be less).
- 48 bits give you 281,474,976,710,656 colours (PNG supports up to this depth).

Usually, when literature refers to 32-bit colour (as on the Apple Macintosh), this refers to 24 bits of colour information and 8 bits of alpha channel or mask, which is used for compositing. Also, a 16-bit system often gives five bits each to red, green and blue, and uses the sixteenth bit for a one-bit mask. Another technique is to give green more bits than red and blue because our eyes are more sensitive to green. The exact format of any particular graphics format will be found in its specification.

The book website includes examples of an image stored with different bit depths.

There are occasions when you will see a reference to colour of more than 24 bits, which is really just that. Colour scanners often scan at 10 or even 12 bits per colour, making 30 or 36 bits. This is because the colour rendition of the object being scanned will not usually match that of the computer, and it is handy to be able to adjust the scanning to compensate for this. The difference usually shows in the darker and lighter parts of the image. This becomes especially true when working with photographic negatives, which have a particularly wide latitude (the film equivalent of dynamic range). The relationship between the colours at the input of a device and its output – for example, as how a monitor displays shades of grey – is called the gamma of the device and is denoted as a number. Every display device has a gamma, and photographs, PCs, television sets and Macs all have different gammas. This means that a 'correct' image made on a Mac will not necessarily look good on a PC or a TV. You can adjust the gamma of an image, but doing this to a 24-bit image results in some colours disappearing – a phenomenon familiar to anyone who uses PhotoShop's 'Levels' facility. Since PNG has 48 bits it will allow changes of gamma as the browser displays an image without sacrificing visible quality. (A good explanation of the maths behind gamma can be found in the PNG specification in the references at the end of this chapter.)

Of course you can substitute 'shade of grey' for colour. It is generally assumed that the eye can distinguish fewer than 256 shades of grey, which would suggest that you could represent a smooth gradation from black to white with 256 shades. Unfortunately this does not always work, because our vision is extremely sensitive to the transition between very similar shades. However, 256 grey levels are reckoned to be far and away good enough, and it certainly works for everyday images. For colour images the boundary lies between 16-bit and 24-bit.

Although 24 bits are best for a colour display, you can make do with fewer bits if the individual colours are chosen carefully from a 24-bit palette of colours, to match the colours in the image. Very few images run across the whole 16 million colours, and so using a custom (or adaptive) palette, as this is called, is a useful way of getting the bit depth, and also the size of the file, down. Often you can go down to 256 colours with a custom palette and the results will be virtually indistinguishable from a full 24-bit image. Unfortunately a custom palette is not always a practical idea, especially if several images with different custom palettes, and even the windows on the screen, are expected to share the same screen display and the screen is not a 24-bit one. The result will, at best, be an attempt by the operating system to reconcile the different palettes but, at worst, some of the images will turn psychedelic. There is a well-known palette for web design that contains the 216 colours 'native' to Netscape Navigator under Windows as a kind of 'lowest common denominator'.
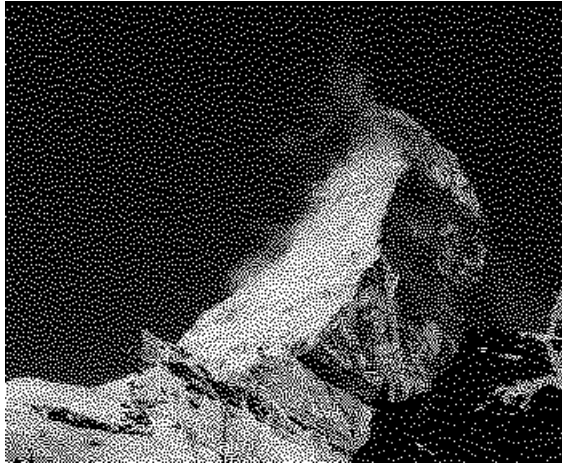
Dithering.

The basic rule of reproduction, and this applies to print as well as to computer displays, is that you can trade spatial information for bit-depth information, which you could call colour resolution. This means that you can group pixels together and use the group, averaged between the individual pixels in the group, to show more colours than an individual pixel can. The group of pixels acts like a larger pixel that is capable of showing more colours. This technique has versions such as half-toning and dithering, which you can use as well as or instead of a custom palette. It is even possible to dither a one-bit image to make it look almost photographic, if the screen resolution is high enough and you do not look too closely.

As an extreme example, the illustration shows a grey-scale image of the Matterhorn that has been dithered from eight bits (256 shades of grey) to one bit (black and white). This is called a diffusion dither. You really have to squint and imagine really hard to see this as a grey-scale image, but the basic principle of dithering is there: groups of dots simulating greys. If you are trying to use a photographic image on the display of a mobile phone then this kind of dither is what you will probably need to do.

■ Using what, when and why

For the designer of a computer-based application there are times when you will need to think very carefully about how your images are stored and reproduced.

In an ideal world, apart from special effects, you would always be able to use 24-bit (or better) images in your web pages and multimedia applications. However, you could run into problems in doing this, for reasons of

display incompatibility and file size and/or download times. If the display of your target delivery system is not 24-bit then you need to know how it will react when fed a 24-bit image. If the display does not gracefully degrade it by, for example, dithering it, you might be better off dithering the images yourself. If there are palette problems, which are common in applications using windowing interfaces because of differing requirements of the windows on the screen, this can make your images look poor to the point of being psychedelic when their window is not active.

Custom palettes, especially when viewed on a display with less than 16-bit colour, present a potential difficulty in that, when switching from one picture to another in a sequence, the palette may not switch at the same time as the picture. This can lead to a brief flash of weird colours on the screen as the palette settles down. There is a brief moment while the beam scanning the image onto the monitor switches off as the display scan moves from the bottom of the screen back to the top, called the vertical blanking interval or VBI and this happens with all monitor displays including televisions. The VBI allows a short period during which both screen image and palette can be changed so that the palette flash is not seen – as long as the necessary changes can be made quickly enough.

File size is important for reasons of capacity and loading time. As well as taking up three times the space of an 8-bit image on your website or CD-ROM, a 24-bit image takes three times as long to load. This could be crucial if you are expecting very fast response times and, of course, this is especially important on a website. As we have already discussed, you can reduce the size of an image by reducing the number of colours, and we discuss data reduction or compression of still images below.

You should also think carefully about whether you want to use 24-bit if it is available. It may be that some of your images will work very well in 8-bit, and so you should consider using them this way. This is particularly true of animations, where you might need to reap the benefit of smaller file sizes, or even use simple but fast compression methods (run-length encoding, for example) to avoid having to move too much data too quickly.

Even though your final delivery may not be a full 24-bit colour image, you should carry out all the retouching and compositing operations in 24-bit. Working on an 8-bit or a 16-bit version of an image will never give as good a result as working on the 24-bit original. This argument also extends to the archiving of graphics, and they should be kept at the highest reasonable quality as well as in their final form.

■ Anti-aliasing

Although text is essentially a one-bit image, there are circumstances when you would want to display text with more bits. It is all part of that trade-off between spatial and colour resolution. When converting from an object (such as text or a drawing) to a bitmap you have to fit the edges of the object

to the pixels on the screen. If you do this and set the pixels black or white depending on whether the majority of the pixel is covered by the line or not, you get jagged lines on the screen, known as jaggies or staircasing.

The alternative is to colour the pixels along the edge in different shades according to how much of each one is covered. This will disguise the jaggies, and usually renders them inconspicuous at the risk of the edges seeming a bit soft. The shades are blends of the foreground and background colours.

Because the effect of the mismatch between the real line and the 'quantizing' caused by trying to fit it onto the pixels in the image is known as aliasing, the kind of rendering that puts the grey pixels around the edge to smooth it out is known as anti-aliasing. Anti-aliased text, particularly, has become very popular with computer typographers and designers because it looks more like words on paper and less like a screen bitmap.

Some computer operating systems can render text to the screen in an anti-aliased fashion in real time, but in other cases the text needs to be built into the image as a graphic.

There are some occasions when you should think carefully about using anti-aliased text. This is especially the case when the text is small, because the grey wedges around the edge tend to make small text look fuzzy. This problem increases with the reducing of text size because the 'grey' pixels around the edge have a fixed size and so become a bigger proportion of the
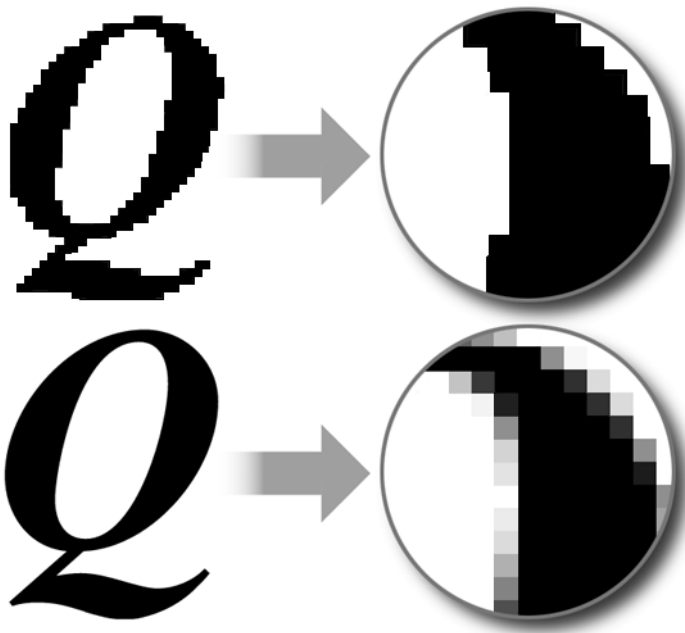
Anti-aliasing.

text size as the text shrinks. In some cases, more careful alignment of the text on the screen will optimize the appearance. The worst-case scenario is where the text is small enough, and misaligned enough, for some of the strokes of the characters to be almost completely grey. In this case they just look out of focus. Making the size of the characters such that their top and bottom edges lie exactly at pixel boundaries reduces this effect, but few characters have straight lines at their edges. Basically you have to check the images carefully.

■ Look before you leap

Any graphic should be checked on the delivery system. Reducing the bit depth of an image can have all sorts of undesirable side-effects, particularly quantization. This is sometimes also called posterization or contouring. It is the result of taking a smooth transition from one colour to another and replacing that with a series of discrete steps.

In this image, as an example, you see an extreme case of posterization. There is a risk that the image on the left (subject to the vagaries of repro-

Posterization.

duction in print), which has 256 shades of grey in it, would come out looking like the one on the right if that computer screen could display only four levels of grey.

Unfortunately you can only be sure of this kind of compatibility if you have checked your image on every screen format, which is a bit problematic for a website because there are so many computers with so many browsers and even Web on TV to consider. As I've already mentioned, there is a well-known group of colours, known as the web safe palette, which can be used for 8-bit images to minimize the risk of unforeseen changes. This palette also eases cross-platform compatibility between the PC and Mac. However, there is a final Mac/PC issue that often catches people out, and that is the difference in the display gamma between the two platforms. As has been discussed earlier in this chapter, gamma is the way in which brightness levels are shown on the display, and the result of Macs and PCs having different gamma curves is that images look darker on a PC than they do on a Mac. To overcome this you must either tailor the image to the platform – a CGI program or JavaScript can find out what platform a viewer's web browser is using and call up appropriate image files – or accept dark or light images or attempt to compromise. In the future it will be possible to detect the characteristics of the computer display and adjust the gamma of a PNG image accordingly.

☐ Taking less space

You will usually need to reduce the size of graphics files. This can be for reasons of space, for example on a compact disk. It can also affect the time it takes to access the files, whether from the Web or from a CD.

One way of reducing the size of a graphic is by degrading it. The size of the file, in pixels, can be reduced. A 640 by 480 file could be shrunk to 320 by 240 and blown back up on display. This will lead to a dramatic reduction in quality. The colour depth of the picture can be reduced. One quite effective way is to change from a 24-bit image to an 8-bit one with only 256 colours. If the palette for the picture is carefully chosen a 256-colour version of a 24-bit picture can be a very effective substitute. This is known as an adaptive palette.

JPEG compresses photographic images substantially by losing the less visible elements of the picture using a process called a discrete cosine transform (DCT) to identify fine detail and remove it. JPEG at relatively low compression rates, such as 5 to 1, will usually be indistinguishable from the original. This means that JPEG is even more efficient than 8-bit with an adaptive palette.

Compare the two images on the next page.

The first image was compressed using a high-quality JPEG compressor, but if we subtract this image from the original, and the range of brightness in the result is expanded so that the changes are easier to see, a mosaic pattern becomes visible.

JPEG Matterhorn.

JPEG artefacts.

In the second image, the finer the dots are, the more detail has been removed by the JPEG DCT algorithm. The reduction achieved by JPEG here is a factor of 3. If the image was colour the difference would be greater. You can find these images on the website.

Reducing the size of a file reduces the time it takes to load that file. If the image needs to be decoded, as a JPEG image would, then you have to take

account of the decoding time. Despite this, on a website you will find that the time taken to download and decode a JPEG image is still less than the time taken to download an uncompressed version – if your browser could display the uncompressed file type.

■ Animated graphics on the Web

Animated graphics are somewhat different from movies, although an animation can be a movie. By animation what we mean is an animated line drawing, like a cartoon, rather than something that is, or looks, photographic. Some forms of animation can be efficiently compressed for use on the Web. We have already mentioned Flash, which takes draw object shapes and text and allows sophisticated animations to be made using them that require very little data. Animated titles or bullet point slides can be made this way with little data overhead over and above the web page itself. The only disadvantage is that the viewer needs to have downloaded the relevant plug-in for the browser and to have installed it correctly otherwise an error results. All browsers can display animated GIF files, so this is an option although, as usual, you need to be careful about download times.

Another form of animation that is web-friendly is sprite animation. A small animated object, called a sprite, can be moved around a larger stage. This sprite could be an animation of a walking figure, and the stage could be a line representing a road. Shockwave, which is based on Macromedia Director, allows animations of this kind to be made and distributed on a web page. Director itself animates by using a sprite-based system, which translates well to the Web. Again, the plug-in is required.

Of course you could devise an animation that runs as a Java applet, and in this case it could be as sophisticated as your programming skill allowed. No plug-in is needed because many browsers support Java, but quite a few users disable it and some companies restrict or ban its use for security reasons.

■ Which graphics format should I use?

To cut to the chase, you have two reasons why one format might be better-suited than another. First, the end-user's browsers may or may not support it. At the time of writing PNG is supported in newer browsers but you can't assume people out there are using them in sufficient numbers. That will change quickly.

Second, the way the image is compressed lends itself to a particular kind of image. GIF is a format that can only handle 256 colours and uses run-length encoding (RLE) compression. This is most efficient when dealing with repeating patterns or large areas of flat colour (because RLE works best

with repeating sequences of identical pixels). So things like company logos and cartoons are usually suitable for GIF.

JPEG is designed for photographs but as these have textures, smooth soft edges and a somewhat arbitrary notion of which parts of the image are sharp and which are soft, you can also use JPEG for any image which does not have hard sharp lines in it, such as a watercolour (or an imitation watercolour) or an air-brush painting.

PNG is still something of a dark horse (at least for most of us). Since it has no lossy compression it will not compress as much as JPEG so it will probably be used as a direct higher quality and more versatile alternative to GIF.

Because GIF and PNG compress an their images with no loss (so you can exactly recreate the original on decompression) you don't really have any trade-off between compression and quality whereas this is a basic choice with JPEG which is a lossy compression standard. However, you can reduce the range of colours in a GIF or PNG and if you can produce a result which looks good then you have successfully reduced the size of the file. Graphics tools for preparing web graphics will usually let you compare versions of the image with different levels of compression. There is also a useful JPEG and GIF fine-tuning tool available on the Net Mechanic website, called GIFBot.

■ Asset management

It is possible that the number of graphics files you might have in your application is large. It could be the largest number of files, and if you take into account versions, sections of composite images, and animations then the number can easily run into thousands.

For this reason it is vital that from the beginning you adopt a known system of naming files. This was especially difficult with filing systems that did not allow long names, such as DOS, but is less of a problem now (although don't be tempted to use very long names since you are likely to run up against a limit of 31 characters on Windows CDs and Macs). It is tempting to use paths in your directory structure to identify files, but this has risks since files can become separated from their directories. Some kind of coding is probably going to be the only safe solution to this problem, with a short code identifying the image and its status. A suffix or extension should be reserved for the file type since some filing systems recognize files only by this extension. The resources in Apple files are potentially very useful, and very detailed information about a file can be put there as a special resource, but this does not help you if you are working on a PC. Also you will have to consider the problems of recognizing files from lists, including lists in archives.

In particularly complex cases a database may be required to handle the assets in an application, and several software companies have been address-

ing the problems of handling an image database of this kind. You may wish to write a system of your own, which can be used for different applications.

■ Image compatibility and quality

Since it is possible to display a photographic-quality image on a 24-bit display, you will need to take care that any compression does not noticeably degrade the image. In agreeing quality standards with a client you will need, as with audio and video, to align your quality with what the delivery system can display. However, there may be times, such as with a website, when the delivery system will encompass a range of displays, and under these circumstances you will need to check how the images look on all the platforms and try to make sure that they degrade gracefully as the display capabilities decline.

From a practical point of view the days of image standards for still pictures being incompatible are almost over, since there are a number of good conversion packages that can take an image in one format and convert it to another with no loss. You should be aware, however, of differences between uncompressed, compressed but lossless, and compressed with loss (or lossy).

An image that is compressed without loss, such as one compressed with run-length encoding, will be the same as an uncompressed one when the latter is expanded. Such a system can be used for archiving and a key example is PNG. Lossy systems are more unpredictable but the good ones, such as Photo-CD and JPEG, can be used as if they were uncompressed as far as delivery of the image is concerned. (There is a lossless JPEG standard but it is rarely used.)

There is a caveat. If you are going to manipulate an image you should start from an uncompressed or lossless-compressed image. Exceptionally, Photo-CD can be used for origination, as the compression system used for it is visually lossless, even though it does lose some information. This loss amounts only to information that we do not see since our eyes do not resolve colour as well as the brightness information.

■ Proofreeding!

In software they say that 'there is always one more bug', and the typo is the graphical equivalent. It will be important to check for any errors in the graphics, just as you will do with any voice-over script. Any text on screen should be checked for spelling and consistency of presentation before it is passed on to the graphics team for display on the screen. However, even the most experienced typographer can make a mistake, and if proper names or foreign languages are involved, extra care is needed.

Proofreading is a skill and, in any event, should not be carried out by the person who wrote the text because they will sometimes see what they expect to see rather than what is really there. You cannot rely on spell-checkers because spell checking does not usually pick up misused words or typos that are valid words, such as 'if' for 'of' or 'reed' for 'read' and in any case a spell checker can't check text in a graphic. But it is still worth running a spell-checker to catch obvious mistakes, missed spaces and such like. You do remember that there was a printing of the Bible which contained the commandment 'Thou shall commit adultery' don't you?

THEORY INTO PRACTICE 8

The website contains examples of images with different resolutions and bit depths. You should familiarize yourself with the effects on different kinds of material and, using whatever image manipulation software you have, experiment with changes in the number of colours.

If you have access to a range of machines with different graphics capabilities, or if you can easily change the display on your computer, you can take a couple of graphic-oriented websites and see how they look under differing conditions.

The GIFBot feature on Net Mechanic (see below) can also be used to directly compare compression options for JPEGs and GIFs.

■ Summary

- There are two basic types of image, one that is object-oriented and known as a drawing or vector graphics, and can be changed relatively easily, and one that consists only of the exact dots displayed on the screen. This is a bitmap or painting, and is less easily manipulated.

- Working with a composite image in a package that uses layers and alpha channels can give you the best of both worlds because you can separate the sections of the image and work on them independently.

- Unless you are working with photographic images (usually 24-bit colour), you will have to select a range of colours carefully for representation of the image. This is known as the palette.

- In transferring a 24-bit image to a smaller palette, say 8-bit, there are compromises that have to be made. However, the resulting file will also be smaller.

- Reducing the number of colours in an image is not the only way of shrinking the file, since JPEG compression, which is an international standard, can achieve a shrinkage with less visible artefacts.

- Whatever you do with images, it is vital always to check how the image will look when it reaches your end-user. And don't forget to proofread any text.

■ Recommended reading

Foley J.D., van Dam A., Feiner S.K. and Hughes J.F. (1996). *Computer Graphics, Principles and Practice*, 2nd edn. Reading, MA: Addison-Wesley

Siegel D. (1997). *Creating Killer Web Sites*, 2nd edn. Indianapolis, IN: Hayden Books

The PNG specification, which has detailed explanations about gamma and other issues, is at
http://www.w3.org/TR/REC-png.html

Net Mechanic's GIFBot is at
http://www.netmechanic.com/accelerate.htm